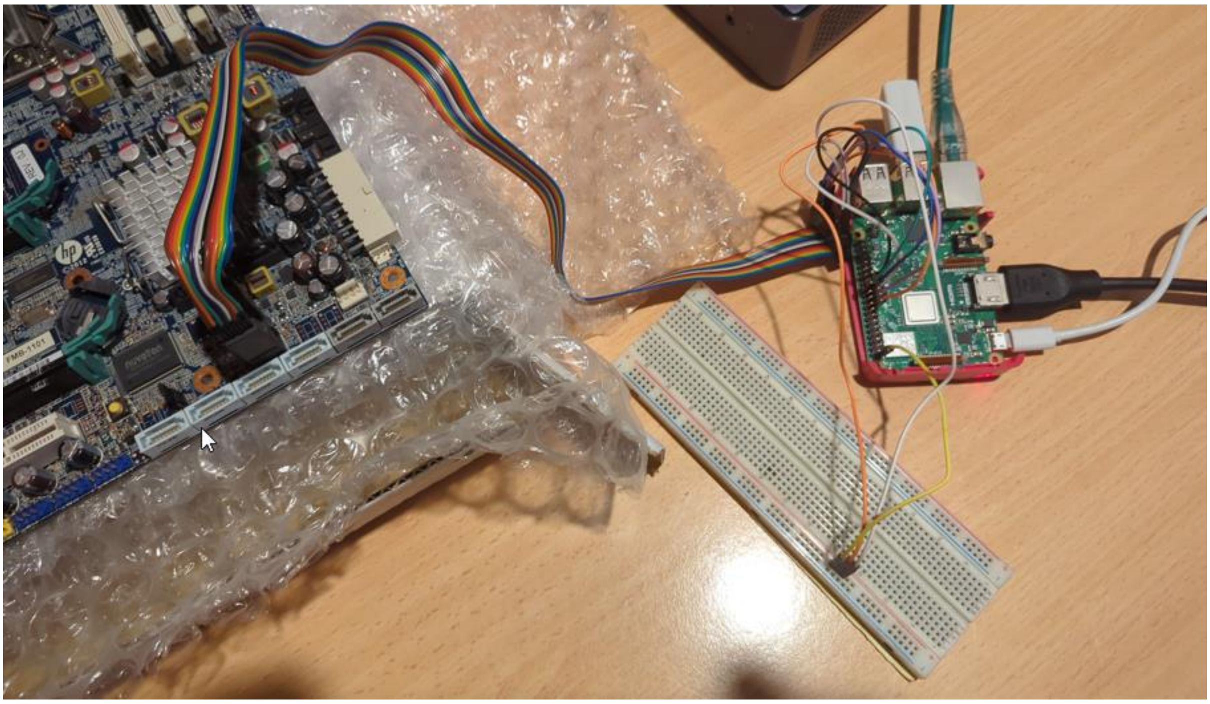
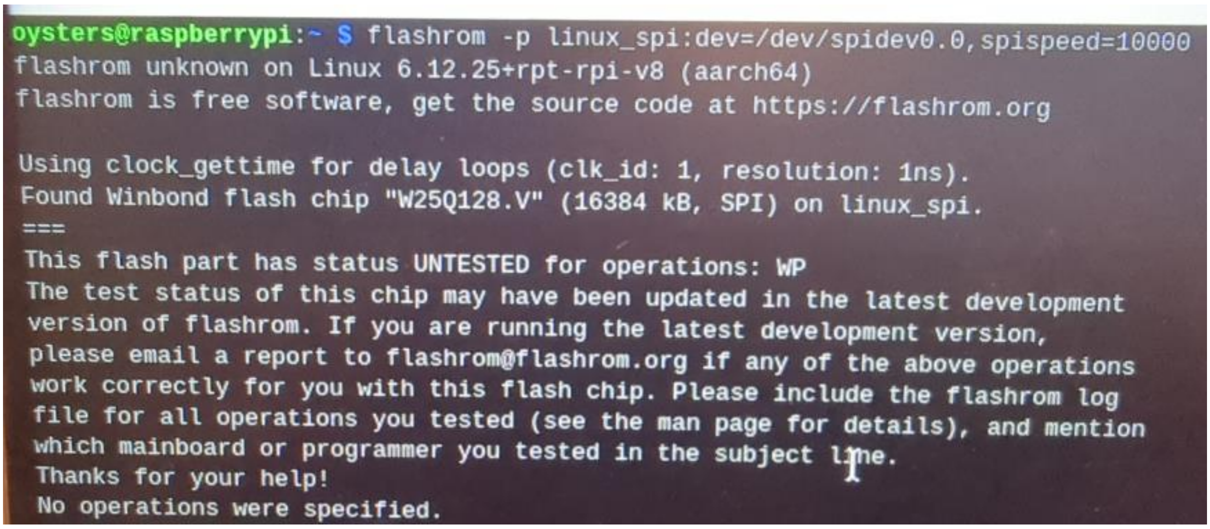
Connection & Reading



Used Rasbperry Pi 3B.

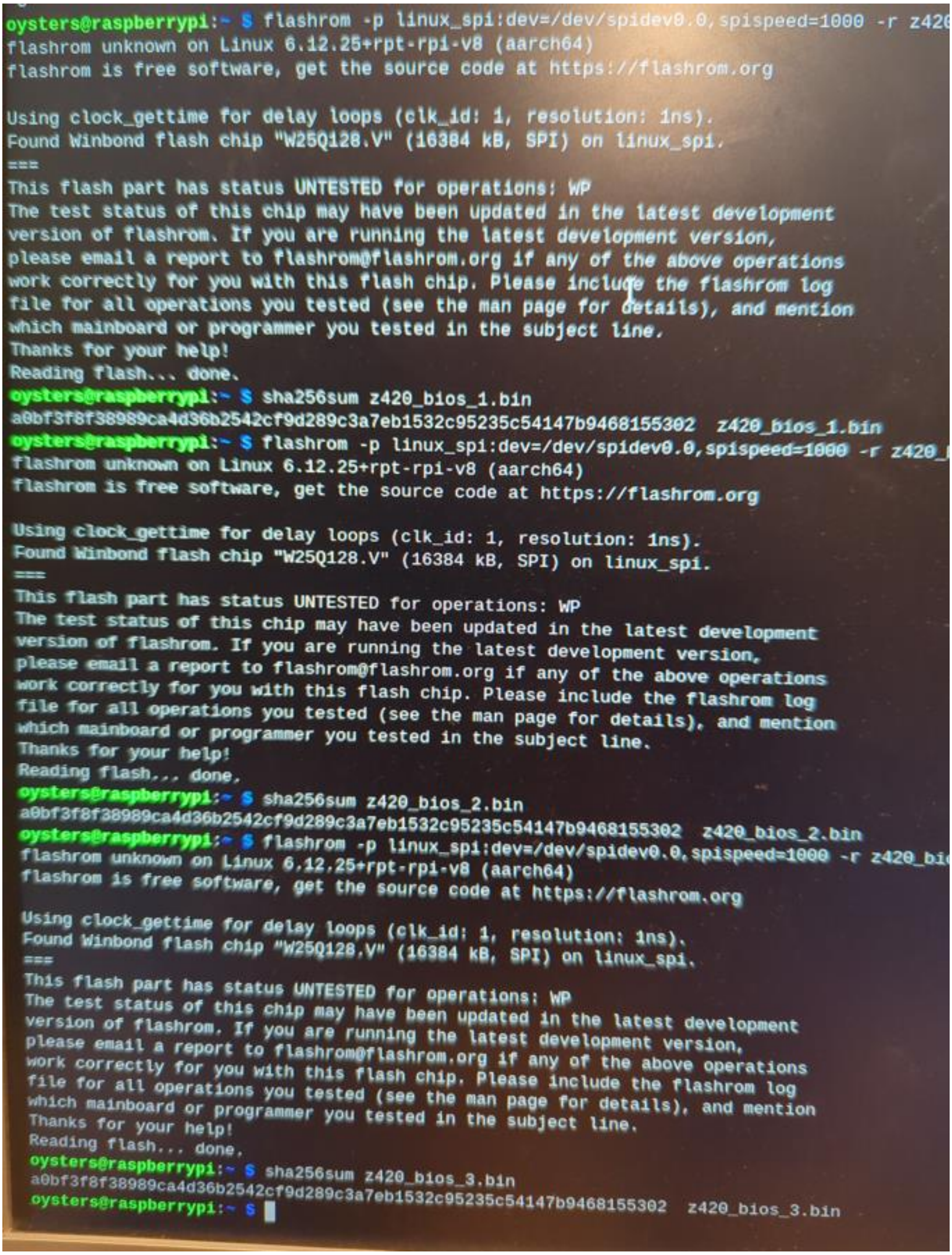
Connectors were mainly M-F for most connections from the plug of the clip (directly connected) and M-M used for power on the breadboard due to not enough pins available on the Pi.

Wiring diagram on SuperThunders write-up. Tried to minimise connections, particularly for data lines.



```
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=10000
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q128.V" (16384 kB, SPI) on linux_spi.
===
This flash part has status UNTESTED for operations: WP
The test status of this chip may have been updated in the latest development
version of flashrom. If you are running the latest development version,
please email a report to flashrom@flashrom.org if any of the above operations
work correctly for you with this flash chip. Please include the flashrom log
file for all operations you tested (see the man page for details), and mention
which mainboard or programmer you tested in the subject line.
Thanks for your help!
No operations were specified.
```

Successful connection will show “Found” and your chip type in this case wasthe standard Winbond.

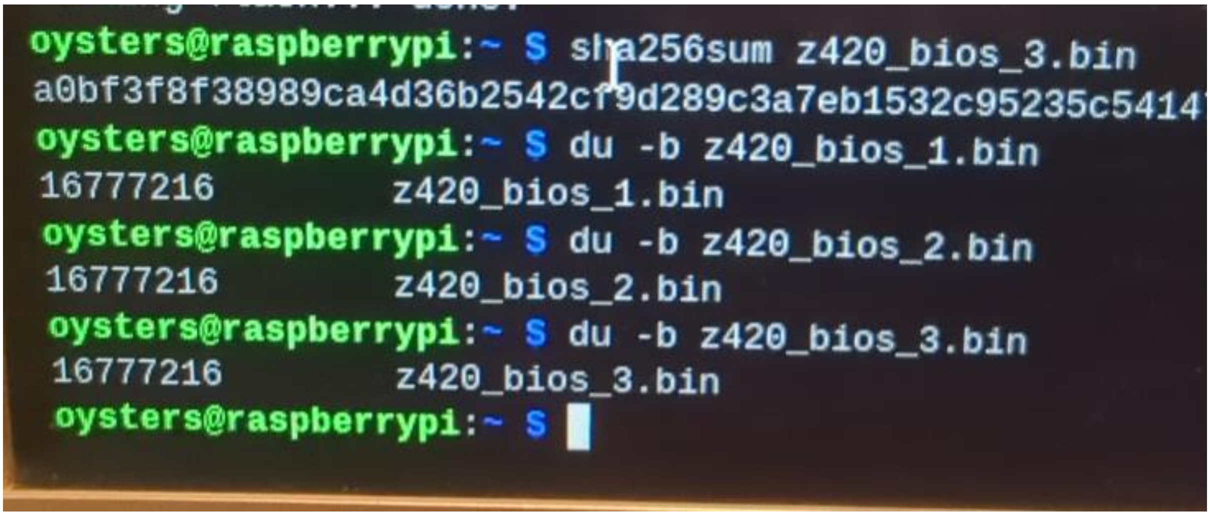


```
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -r z420
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q128.V" (16384 kB, SPI) on linux_spi.
===
This flash part has status UNTESTED for operations: WP
The test status of this chip may have been updated in the latest development
version of flashrom. If you are running the latest development version,
please email a report to flashrom@flashrom.org if any of the above operations
work correctly for you with this flash chip. Please include the flashrom log
file for all operations you tested (see the man page for details), and mention
which mainboard or programmer you tested in the subject line.
Thanks for your help!
Reading flash... done.
oysters@raspberrypi:~ $ sha256sum z420_bios_1.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_1.bin
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -r z420_
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q128.V" (16384 kB, SPI) on linux_spi.
===
This flash part has status UNTESTED for operations: WP
The test status of this chip may have been updated in the latest development
version of flashrom. If you are running the latest development version,
please email a report to flashrom@flashrom.org if any of the above operations
work correctly for you with this flash chip. Please include the flashrom log
file for all operations you tested (see the man page for details), and mention
which mainboard or programmer you tested in the subject line.
Thanks for your help!
Reading flash... done.
oysters@raspberrypi:~ $ sha256sum z420_bios_2.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_2.bin
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -r z420_bi
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q128.V" (16384 kB, SPI) on linux_spi.
===
This flash part has status UNTESTED for operations: WP
The test status of this chip may have been updated in the latest development
version of flashrom. If you are running the latest development version,
please email a report to flashrom@flashrom.org if any of the above operations
work correctly for you with this flash chip. Please include the flashrom log
file for all operations you tested (see the man page for details), and mention
which mainboard or programmer you tested in the subject line.
Thanks for your help!
Reading flash... done.
oysters@raspberrypi:~ $ sha256sum z420_bios_3.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_3.bin
oysters@raspberrypi:~ $ 
```

Same hash values from 3 reads.

Checking

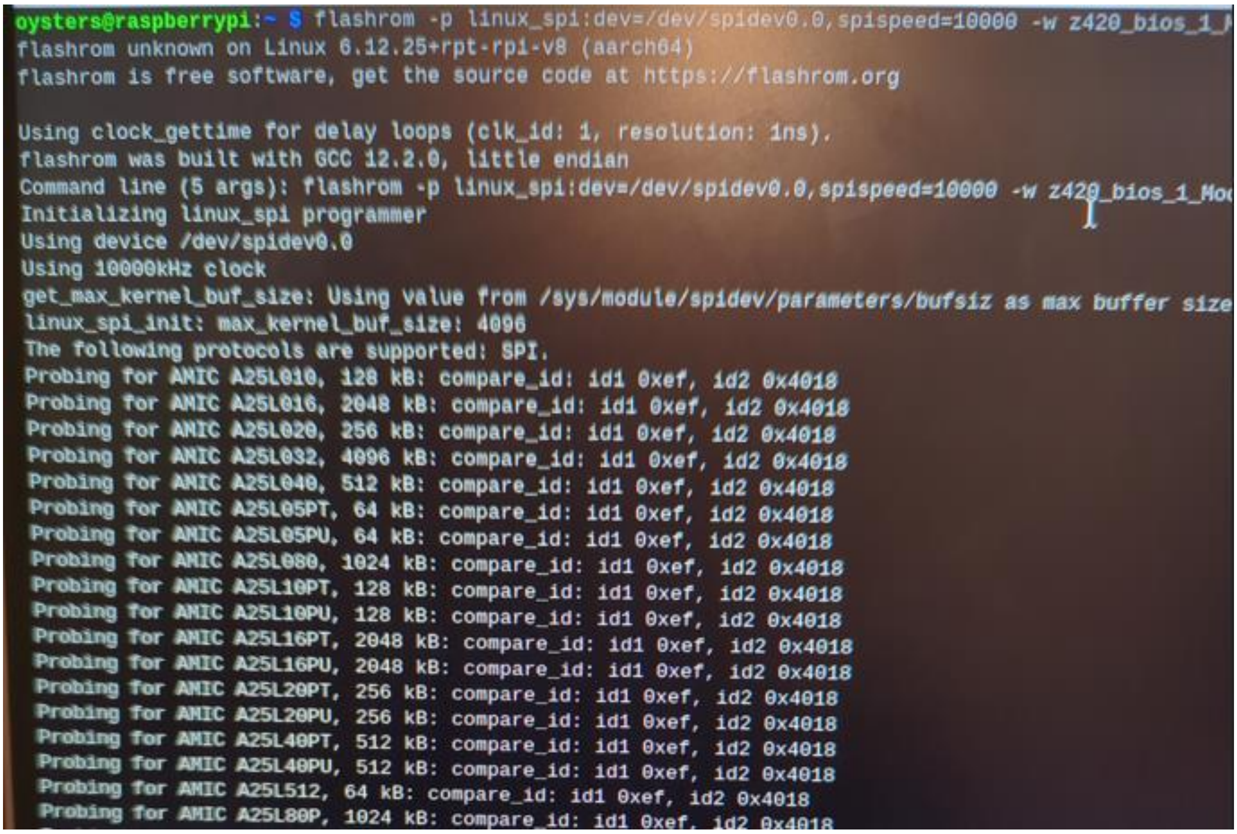
Completed 3 downloads from the chip (saved as z420_bios_1, 2, 3.bin) and checked hash value to ensure all are the same.



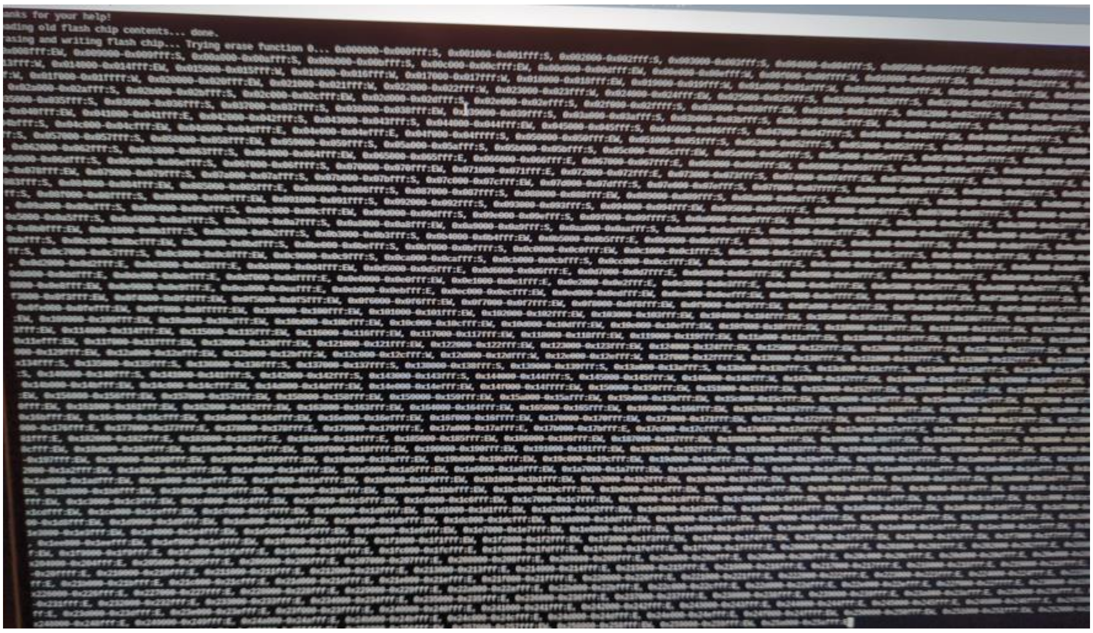
```
oysters@raspberrypi:~ $ sha256sum z420_bios_3.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c5414
oysters@raspberrypi:~ $ du -b z420_bios_1.bin
16777216        z420_bios_1.bin
oysters@raspberrypi:~ $ du -b z420_bios_2.bin
16777216        z420_bios_2.bin
oysters@raspberrypi:~ $ du -b z420_bios_3.bin
16777216        z420_bios_3.bin
oysters@raspberrypi:~ $
```

Checked size, all the same, should be 16777216.

Writing to Chip



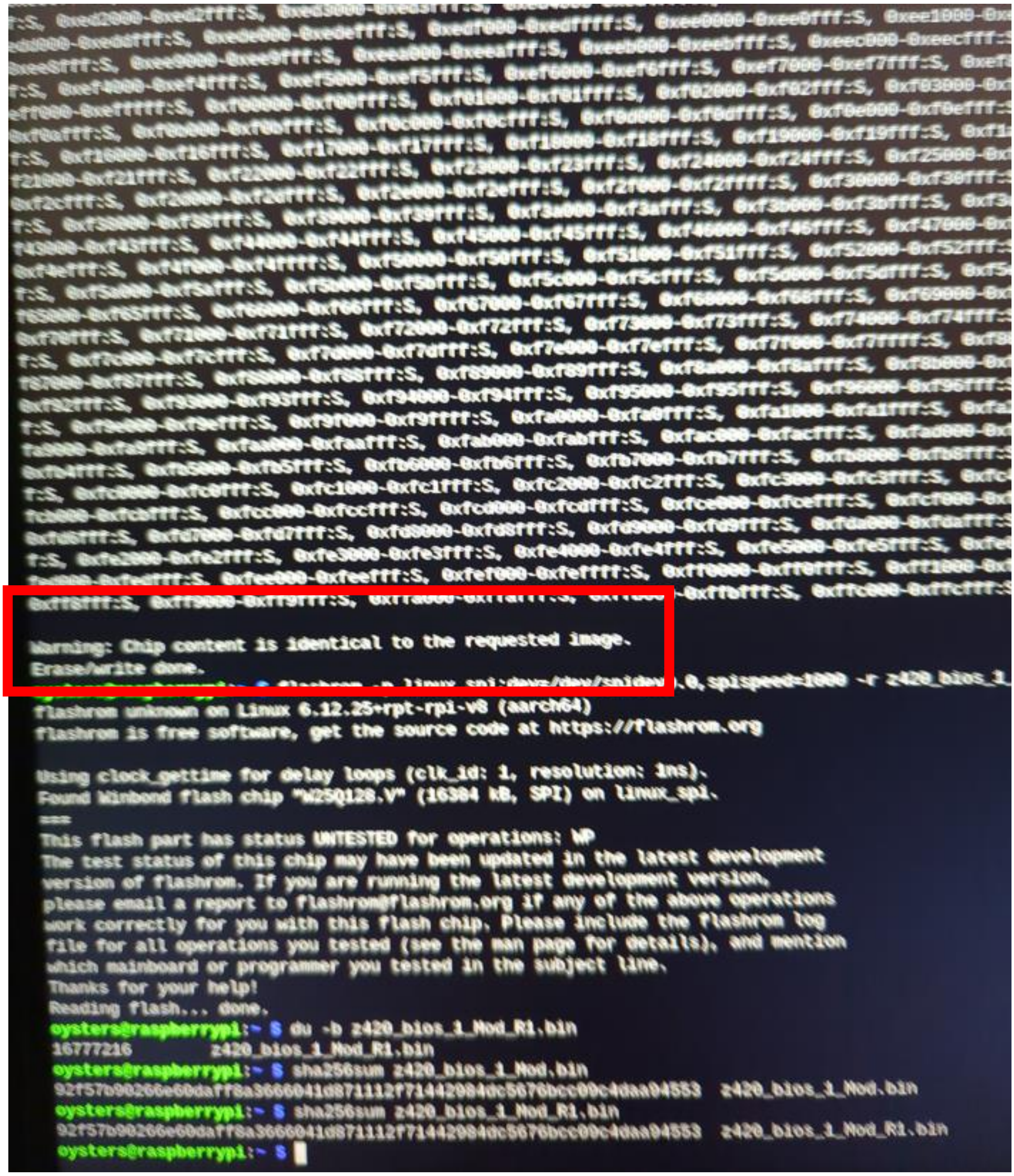
```
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=10000 -w z420_bios_1_M
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
flashrom was built with GCC 12.2.0, little endian
Command line (5 args): flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=10000 -w z420_bios_1_Mod
Initializing linux_spi programmer
Using device /dev/spidev0.0
Using 10000kHz clock
get_max_kernel_buf_size: Using value from /sys/module/spidev/parameters/bufsiz as max buffer size
linux_spi_init: max_kernel_buf_size: 4096
The following protocols are supported: SPI.
Probing for AMIC A25L010, 128 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L016, 2048 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L020, 256 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L032, 4096 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L040, 512 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L05PT, 64 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L05PU, 64 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L080, 1024 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L10PT, 128 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L10PU, 128 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L16PT, 2048 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L16PU, 2048 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L20PT, 256 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L20PU, 256 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L40PT, 512 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L40PU, 512 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L512, 64 kB: compare_id: id1 0xef, id2 0x4018
Probing for AMIC A25L80P, 1024 kB: compare_id: id1 0xef, id2 0x4018
```



```
anks for your help!
ading old flash chip contents... done.
asing and writing flash chip... Trying erase function 0... 0x000000-0x000fff:S, 0x001000-0x001fff:S,
0x008fff:EW, 0x009000-0x009fff:S, 0x00a000-0x00afff:S, 0x00b000-0x00bfff:S, 0x00c000-0x00cfff:EW,
13fff:W, 0x014000-0x014fff:EW, 0x015000-0x015fff:W, 0x016000-0x016fff:W, 0x017000-0x017fff:W,
```

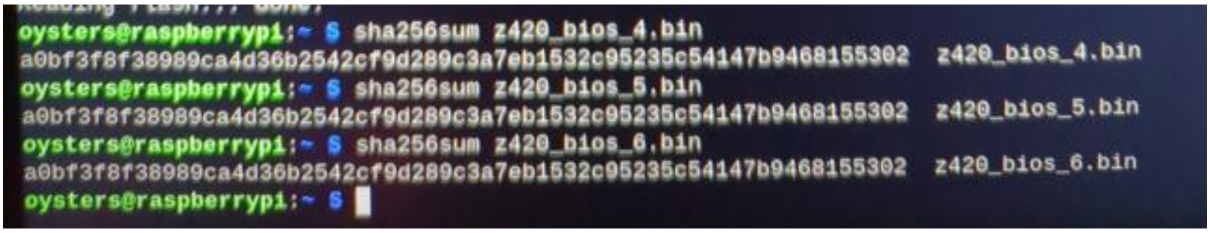
After this it failed to complete, didn’t catch that screen. However after detatching and re-attaching clip to the Bios chip, checking with a read and re-writing it declared: *“Warning: Chip content is identical to the requested image”*

The whole time the raspberry Pi was throwing low voltage warnings despite having large enough power supply. Larger supply was used for the second write which probably helped it finish.



```
f:S, 0xed2000-0xed2fff:S, 0xed3000-0xed3fff:S,
dd000-0xeddfff:S, 0xede000-0xedefff:S, 0xedf000-0xedffff:S, 0xee0000-0xee0fff:S, 0xee1000-0x
0xee8fff:S, 0xee9000-0xee9fff:S, 0xeea000-0xeeafff:S, 0xeeb000-0xeebfff:S, 0xeec000-0xeecfff:S
f:S, 0xef4000-0xef4fff:S, 0xef5000-0xef5fff:S, 0xef6000-0xef6fff:S, 0xef7000-0xef7fff:S, 0xef8
eff000-0xefffff:S, 0xf00000-0xf00fff:S, 0xf01000-0xf01fff:S, 0xf02000-0xf02fff:S, 0xf03000-0x
0xf0afff:S, 0xf0b000-0xf0bfff:S, 0xf0c000-0xf0cfff:S, 0xf0d000-0xf0dfff:S, 0xf0e000-0xf0efff:S
f:S, 0xf16000-0xf16fff:S, 0xf17000-0xf17fff:S, 0xf18000-0xf18fff:S, 0xf19000-0xf19fff:S, 0xf1a
f21000-0xf21fff:S, 0xf22000-0xf22fff:S, 0xf23000-0xf23fff:S, 0xf24000-0xf24fff:S, 0xf25000-0x
0xf2cfff:S, 0xf2d000-0xf2dfff:S, 0xf2e000-0xf2efff:S, 0xf2f000-0xf2ffff:S, 0xf30000-0xf30fff:S
f:S, 0xf38000-0xf38fff:S, 0xf39000-0xf39fff:S, 0xf3a000-0xf3afff:S, 0xf3b000-0xf3bfff:S, 0xf3c
f43000-0xf43fff:S, 0xf44000-0xf44fff:S, 0xf45000-0xf45fff:S, 0xf46000-0xf46fff:S, 0xf47000-0x
0xf4efff:S, 0xf4f000-0xf4ffff:S, 0xf50000-0xf50fff:S, 0xf51000-0xf51fff:S, 0xf52000-0xf52fff:S
f:S, 0xf5a000-0xf5afff:S, 0xf5b000-0xf5bfff:S, 0xf5c000-0xf5cfff:S, 0xf5d000-0xf5dfff:S, 0xf5e
f65000-0xf65fff:S, 0xf66000-0xf66fff:S, 0xf67000-0xf67fff:S, 0xf68000-0xf68fff:S, 0xf69000-0x
0xf70fff:S, 0xf71000-0xf71fff:S, 0xf72000-0xf72fff:S, 0xf73000-0xf73fff:S, 0xf74000-0xf74fff:S
f:S, 0xf7c000-0xf7cfff:S, 0xf7d000-0xf7dfff:S, 0xf7e000-0xf7efff:S, 0xf7f000-0xf7ffff:S, 0xf80
f87000-0xf87fff:S, 0xf88000-0xf88fff:S, 0xf89000-0xf89fff:S, 0xf8a000-0xf8afff:S, 0xf8b000-0x
0xf92fff:S, 0xf93000-0xf93fff:S, 0xf94000-0xf94fff:S, 0xf95000-0xf95fff:S, 0xf96000-0xf96fff:S
f:S, 0xf9e000-0xf9efff:S, 0xf9f000-0xf9ffff:S, 0xfa0000-0xfa0fff:S, 0xfa1000-0xfa1fff:S, 0xfa2
fa9000-0xfa9fff:S, 0xfaa000-0xfaafff:S, 0xfab000-0xfabfff:S, 0xfac000-0xfacfff:S, 0xfad000-0x
0xfb4fff:S, 0xfb5000-0xfb5fff:S, 0xfb6000-0xfb6fff:S, 0xfb7000-0xfb7fff:S, 0xfb8000-0xfb8fff:S
f:S, 0xfc0000-0xfc0fff:S, 0xfc1000-0xfc1fff:S, 0xfc2000-0xfc2fff:S, 0xfc3000-0xfc3fff:S, 0xfc4
fcb000-0xfcbfff:S, 0xfcc000-0xfccfff:S, 0xfcd000-0xfcdfff:S, 0xfce000-0xfcefff:S, 0xfcf000-0x
0xfd6fff:S, 0xfd7000-0xfd7fff:S, 0xfd8000-0xfd8fff:S, 0xfd9000-0xfd9fff:S, 0xfda000-0xfdafff:S
f:S, 0xfe2000-0xfe2fff:S, 0xfe3000-0xfe3fff:S, 0xfe4000-0xfe4fff:S, 0xfe5000-0xfe5fff:S, 0xfe6
fed000-0xfedfff:S, 0xfee000-0xfeefff:S, 0xfef000-0xfeffff:S, 0xff0000-0xff0fff:S, 0xff1000-0x
0xff8fff:S, 0xff9000-0xff9fff:S, 0xffa000-0xffafff:S, 0xffb000-0xffbfff:S, 0xffc000-0xffcfff:S

Warning: Chip content is identical to the requested image.
Erase/write done.
oysters@raspberrypi:~ $ flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -r z420_bios_1_
flashrom unknown on Linux 6.12.25+rpt-rpi-v8 (aarch64)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Winbond flash chip "W25Q128.V" (16384 kB, SPI) on linux_spi.
===
This flash part has status UNTESTED for operations: WP
The test status of this chip may have been updated in the latest development
version of flashrom. If you are running the latest development version,
please email a report to flashrom@flashrom.org if any of the above operations
work correctly for you with this flash chip. Please include the flashrom log
file for all operations you tested (see the man page for details), and mention
which mainboard or programmer you tested in the subject line.
Thanks for your help!
Reading flash... done.
oysters@raspberrypi:~ $ du -b z420_bios_1_Mod_R1.bin
16777216        z420_bios_1_Mod_R1.bin
oysters@raspberrypi:~ $ sha256sum z420_bios_1_Mod.bin
92f57b90266e60daff8a3666041d871112f71442984dc5676bcc09c4daa94553  z420_bios_1_Mod.bin
oysters@raspberrypi:~ $ sha256sum z420_bios_1_Mod_R1.bin
92f57b90266e60daff8a3666041d871112f71442984dc5676bcc09c4daa94553  z420_bios_1_Mod_R1.bin
oysters@raspberrypi:~ $
```

Checking



```
Reading flash... done.
oysters@raspberrypi:~ $ sha256sum z420_bios_4.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_4.bin
oysters@raspberrypi:~ $ sha256sum z420_bios_5.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_5.bin
oysters@raspberrypi:~ $ sha256sum z420_bios_6.bin
a0bf3f8f38989ca4d36b2542cf9d289c3a7eb1532c95235c54147b9468155302  z420_bios_6.bin
oysters@raspberrypi:~ $
```

After flashing (writing), read again 3 times and checked hash was the same.

Opened those bins with bin used to write in HxD and they were the same.